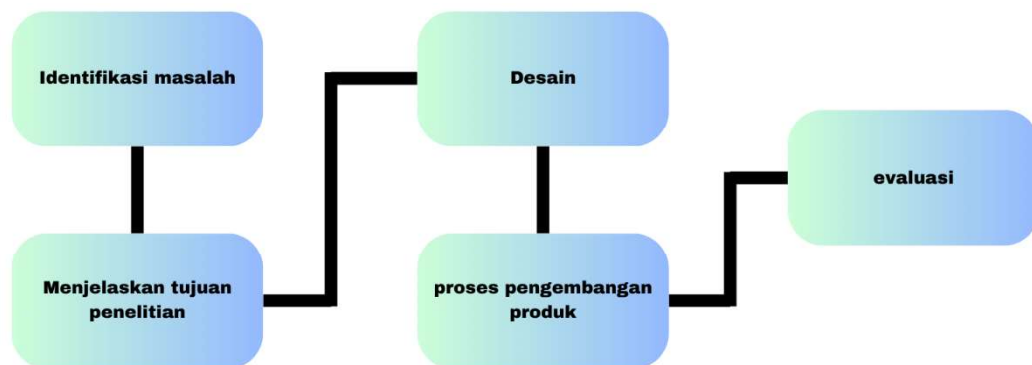


BAB III METODE PENELITIAN

Metode penelitian sangat penting dalam melakukan sebuah penelitian dikarenakan penentuan metode penelitian akan mempengaruhi hasil akhir dari penelitian yang dilakukan salah satu contohnya adalah penyelesaian rumusan masalah yang dirancang atau penyelesaian tujuan masalah. Metode penelitian yang digunakan dalam penelitian ini adalah metode *Design and Development (D&D)*.

Tujuan dari metode penelitian D&D menurut Rich dan Klein adalah membuat sebuah produk baru atau menyempurnakan produk yang sudah ada dengan menambahkan fitur baru atau memperbaiki kesalahan pada penelitian sebelumnya dengan melakukan beberapa proses menurut Pefferes diantaranya mengidentifikasi masalah yang ada, menjelaskan tujuan penelitian, desain, proses pengembangan produk, dan proses evaluasi (Lestari dkk., 2018). Berikut adalah alur penelitian *Design and Development* yang dapat dilihat pada gambar 3.1.



Gambar 3.1 Metode penelitian *Desain and Development* menurut Pefferes

3.1 Identifikasi masalah

Saat ini permasalahan dalam pemberian pakan kucing masih secara manual yang kurang efektif karena dapat menghabiskan waktu dan tenaga. Permasalahan lainnya pemilik hewan tidak sedang berada di rumah dikarenakan bekerja ataupun aktivitas lainnya. Solusi dari permasalahan diatas adalah dengan membuat sebuah pemberi pakan hewan peliharaan secara otomatis yang bisa diatur secara manual ataupun bisa secara penjadwalan menggunakan konsep IoT yang bisa dijalankan di aplikasi android.

3.2 Menjelaskan tujuan penelitian

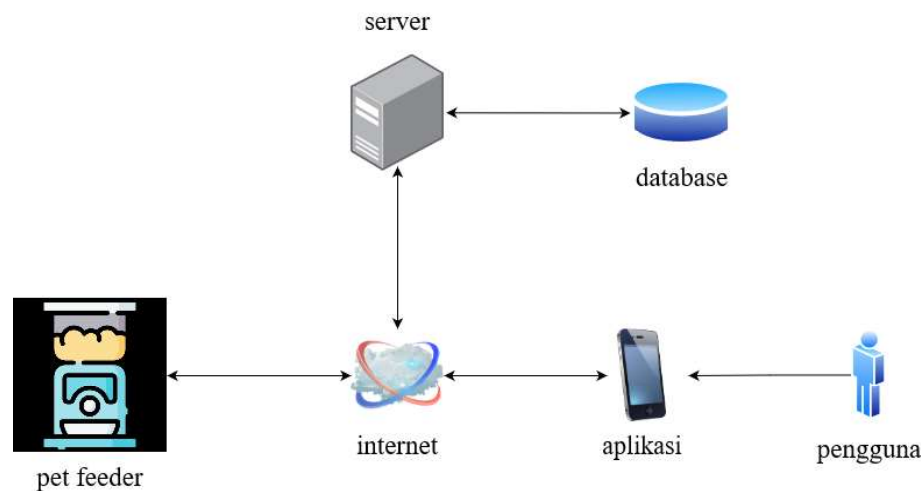
Berdasarkan identifikasi masalah yang telah dijelaskan sebelumnya, tujuan dari penelitian ini adalah untuk memudahkan pemilik hewan peliharaan dalam memberikan pakan hewan peliharaannya yang bisa dikendalikan dari jarak jauh.

3.3 Desain

Merupakan sebuah proses perancangan produk yang akan dibuat nantinya. Proses ini sangat penting dilakukan sebelum melakukan tahapan produksi alat yang dibuat dikarenakan dengan menjalankan proses ini dapat mengurangi kesalahan pada tahap pengembangan serta menjadi petunjuk dalam penelitian agar tidak keluar dari rumusan masalah yang ditentukan sebelumnya. Dalam penelitian ini dilakukan beberapa desain yang dilakukan diantaranya adalah diagram arsitektur, diagram blok, *use case diagram*, ERD diagram, desain 3D wadah *pet feeder*, desain PCB, dan desain tampilan aplikasi android.

3.3.1 Diagram arsitektur

Diagram arsitektur adalah gambaran secara visual terkait sistem yang akan dibuat dengan tujuan lebih terstruktur dan memberikan pemahaman yang lebih jelas serta dapat menjelaskan fungsi dari komponen yang digunakan saling berinteraksi. arsitektur diagram dari *pet feeder* yang dibuat dapat dilihat pada gambar 3.2.



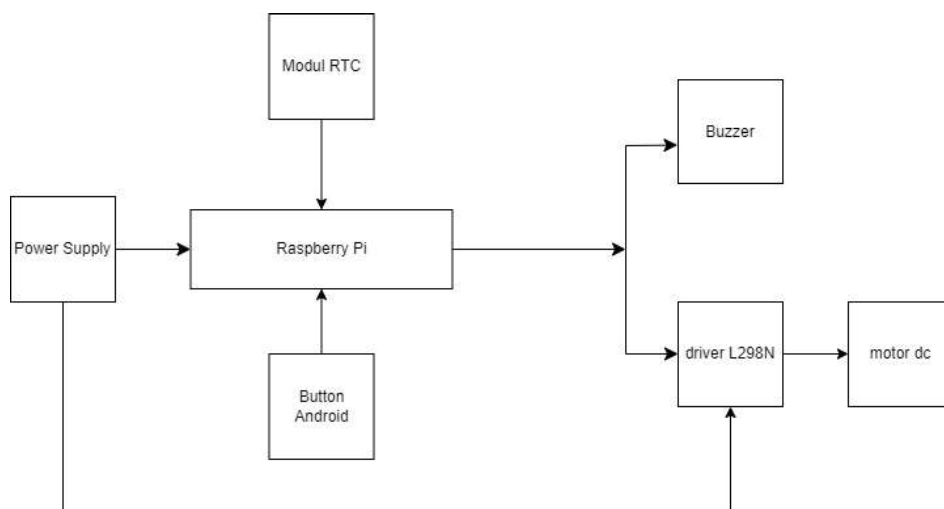
Gambar 3.2 Diagram arsitektur sistem

Diperlihatkan pada gambar 3.2 terdapat beberapa aktor yang pertama yaitu pengguna, lalu terdapat aplikasi yang memiliki fungsi sebagai pengontrol dari *pet feeder* yang dibuat, aplikasi yang dibuat terhubung dengan internet yang berfungsi sebagai sarana untuk bertukarnya informasi antara aplikasi, *pet feeder* dengan

server. Lalu terdapat sebuah *server* yang melayani permintaan data ke *database*. Terakhir terdapat sebuah *database* yang memiliki fungsi menyimpan semua data yang dibutuhkan dalam aplikasi ataupun oleh *pet feeder* nya itu sendiri.

3.3.1 Perancangan perangkat keras

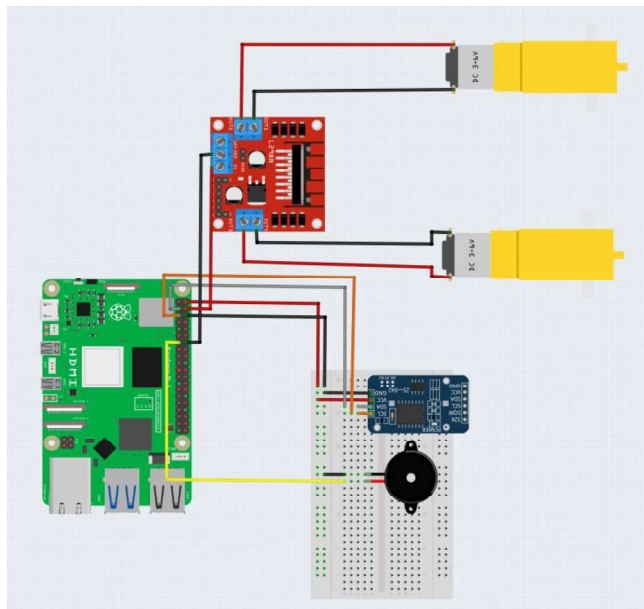
Setelah mengetahui masalah yang akan diselesaikan dan juga sudah mengetahui tujuan penelitian yang dilakukan langkah selanjutnya adalah melakukan perancangan perangkat keras yang diutamakan dapat memberikan pakan kepada hewan secara manual, otomatis dan penjadwalan. Komponen yang digunakan terdiri dari Raspberry Pi 5, motor driver L298N, motor DC, *buzzer*, dan modul RTC. Untuk merepresentasikan bagaimana semua komponen yang digunakan saling berinteraksi dapat dilihat pada gambar 3.3.



Gambar 3.3 Diagram blok sistem

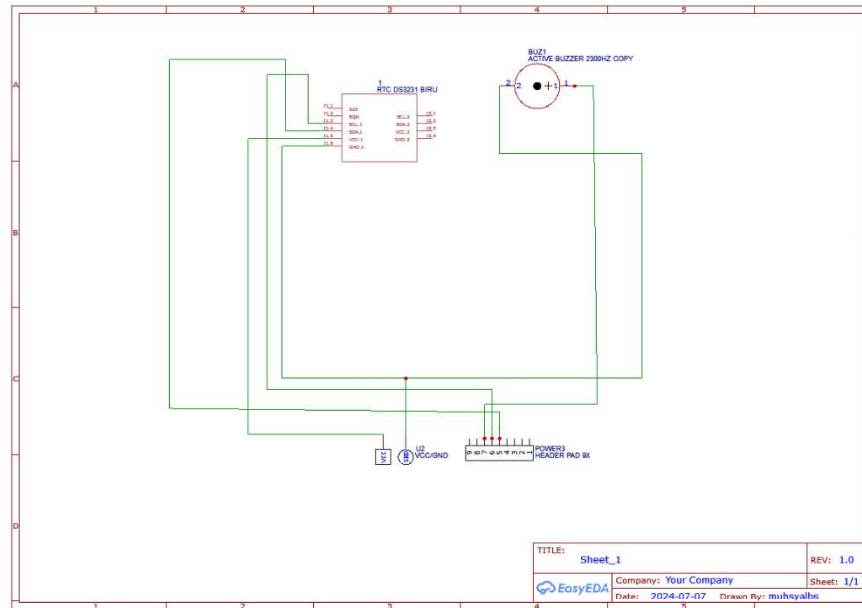
Berdasarkan pada gambar 3.3 diperlihatkan diagram blok sistem yang digunakan dalam penelitian ini. Pertama *power supply* memberikan daya kepada Raspberry Pi dan juga kepada motor *driver* L298N karena dua komponen tersebut merupakan sebuah *single board computer* dan motor *driver* yang nantinya memberikan daya kepada sensor dan komponen lainnya. Lalu disini modul RTC merupakan salah satu input untuk pemberi waktu secara nyata dalam mengatur jadwal pakan kucing, untuk *button android* merupakan sebuah input yang fungsinya dalam pemberian pakan kucing secara manual jika *button* bernilai 1 maka akan mengirimkan data ke *raspberry Pi* untuk menjalankan motor DC.

Setelah membuat blok diagram adalah membuat *wiring diagram* yang memiliki tujuan untuk merepresentasikan secara visual bagaimana komponen yang digunakan saling terhubung mulai dari daya, sensor, actuator dan komponen lainnya serta berfungsi sebagai memudahkan peneliti untuk merancang alat yang akan dibuat. Untuk *wiring diagram* dapat dilihat pada gambar 3.4.



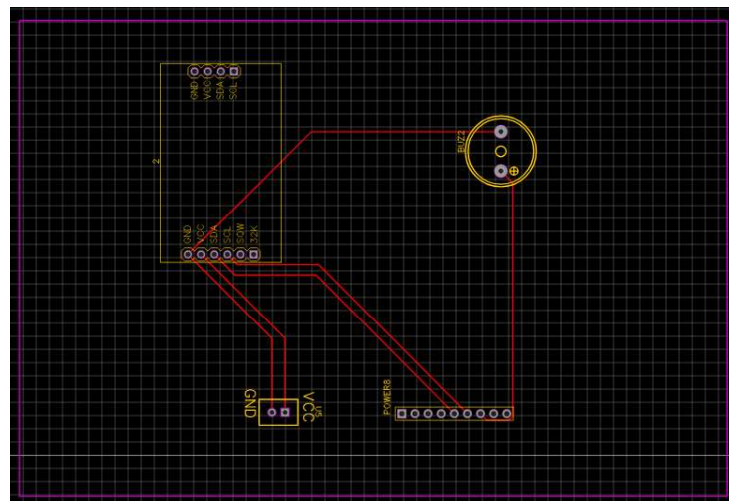
Gambar 3.4 *Wiring diagram* keseluruhan alat yang dibuat

Pada gambar 3.4 diperlihatkan *wiring diagram* dari keseluruhan komponen yang digunakan seperti raspberry pi 5, motor driver L298N, motor DC, modul RTC, dan juga *buzzer*. Setelah membuat *wiring diagram* adalah membuat *schematic diagram* yang memiliki fungsi yaitu untuk memvisualisasikan sebuah rangkaian *pet feeder* yang dibuat. Dalam penelitian ini menggunakan *schematic diagram* sebagai dokumentasi perakitan dalam sebuah PCB yang nantinya akan dibuat agar tidak terjadi kesalahan. *Schematic diagram* dapat dilihat pada gambar 3.5.



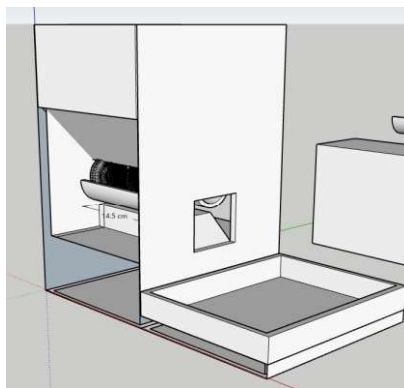
Gambar 3.5 Schematic diagram PCB

Setelah membuat *schematic* diagram adalah membuat desain PCB (Printed Circuit Board) dengan tujuan agar komponen yang digunakan dapat diorganisir lebih rapih dan juga digunakan sebagai pengganti kabel *jumper* yang biasanya untuk menghubungkan antar komponen baik penghubung daya ataupun penghubung ke pin di mikrokontroler. Disini terdapat desain PCB yang peneliti buat yang dapat dilihat pada gambar 3.6

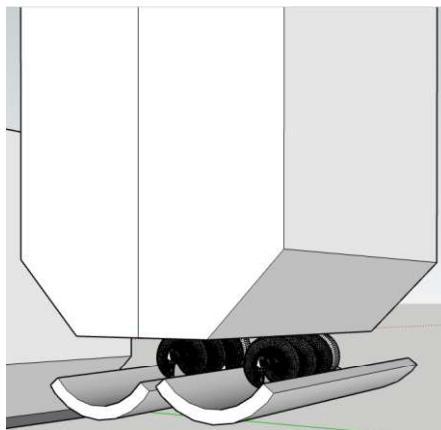


Gambar 3.6 Desain PCB yang dibuat

Setelah membuat desain PCB adalah membuat desain 3D wadah *pet feeder* untuk menyimpan beberapa komponen yang digunakan serta menyimpan pakan hewan peliharaan. Fungsi dari pembuatan desain 3D *pet feeder* ini yaitu memberikan sebuah gambaran *pet feeder* yang akan dibuat serta meminimalisir kesalahan dalam pembuatan wadah *pet feeder*. Terdapat beberapa bagian yang dibuat, pertama dari wadah *pet feeder* yang berbahan kayu serta untuk mekanisme pengeluaran pakan menggunakan filamen. Untuk desain 3D *pet feeder* dapat dilihat pada gambar 3.7 dan gambar 3.8

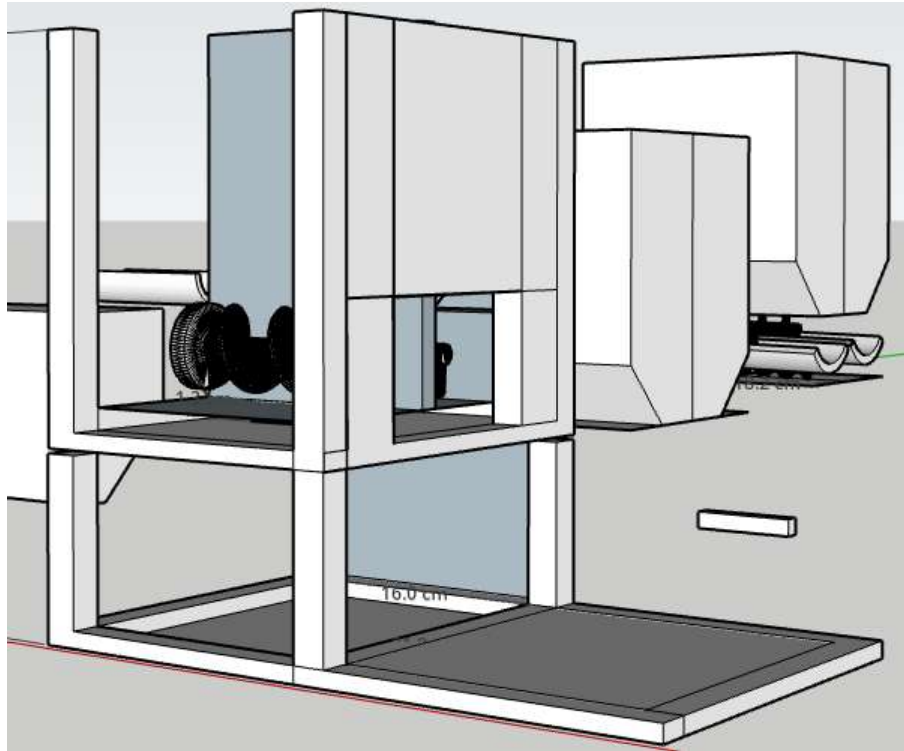


Gambar 3.7 Desain 3D wadah *pet feeder*



Gambar 3.8 Desain 3D mekanisme keluarnya pakan

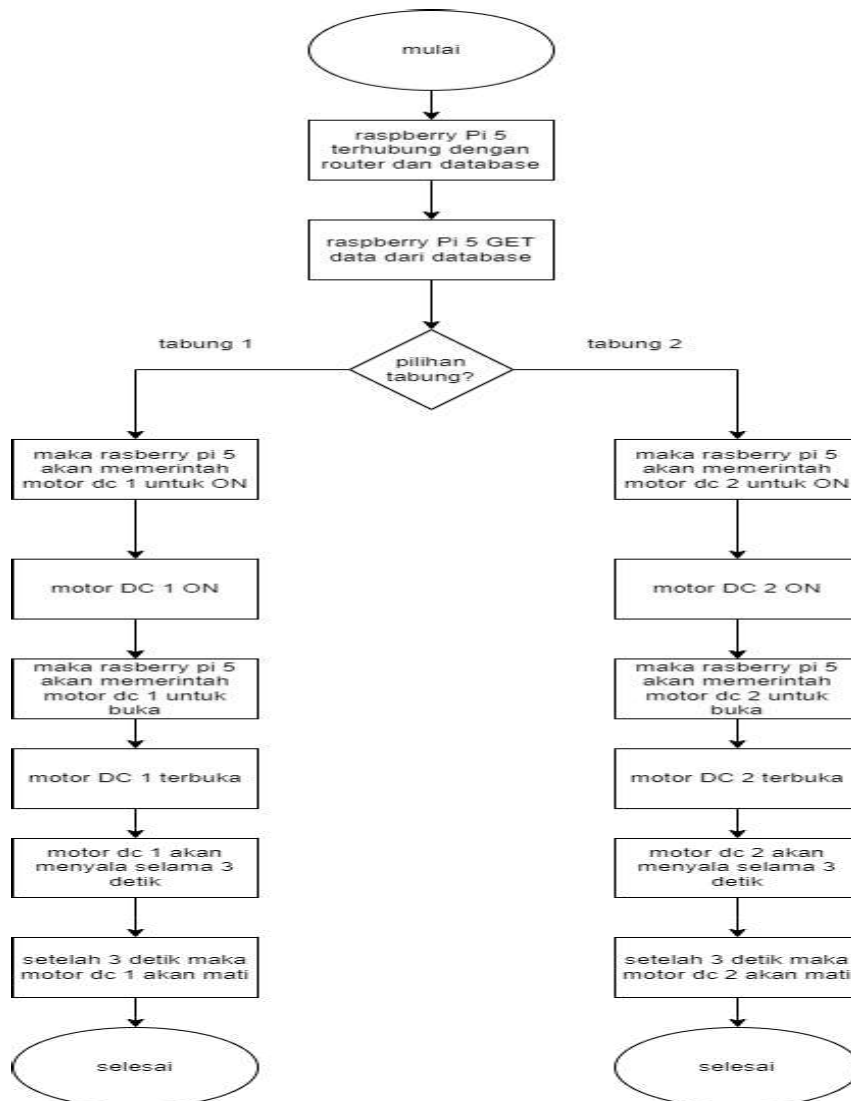
Pada gambar 3.8 diperlihatkan sebuah mekanisme keluarnya pakan yang berbentuk spiral sebanyak dua buah. Tujuan pemilihan mekanisme berbentuk spiral ini adalah agar pakan yang dikeluarkan tidak berhamburan. Selain itu, untuk memperkuat dan memperkokoh wadah *pet feeder* yang dibuat maka dibuatkan sebuah kerangka berbahan dasar kayu seperti yang ditunjukkan pada gambar 3.9.



Gambar 3.9 Kerangka untuk wadah yang dibuat

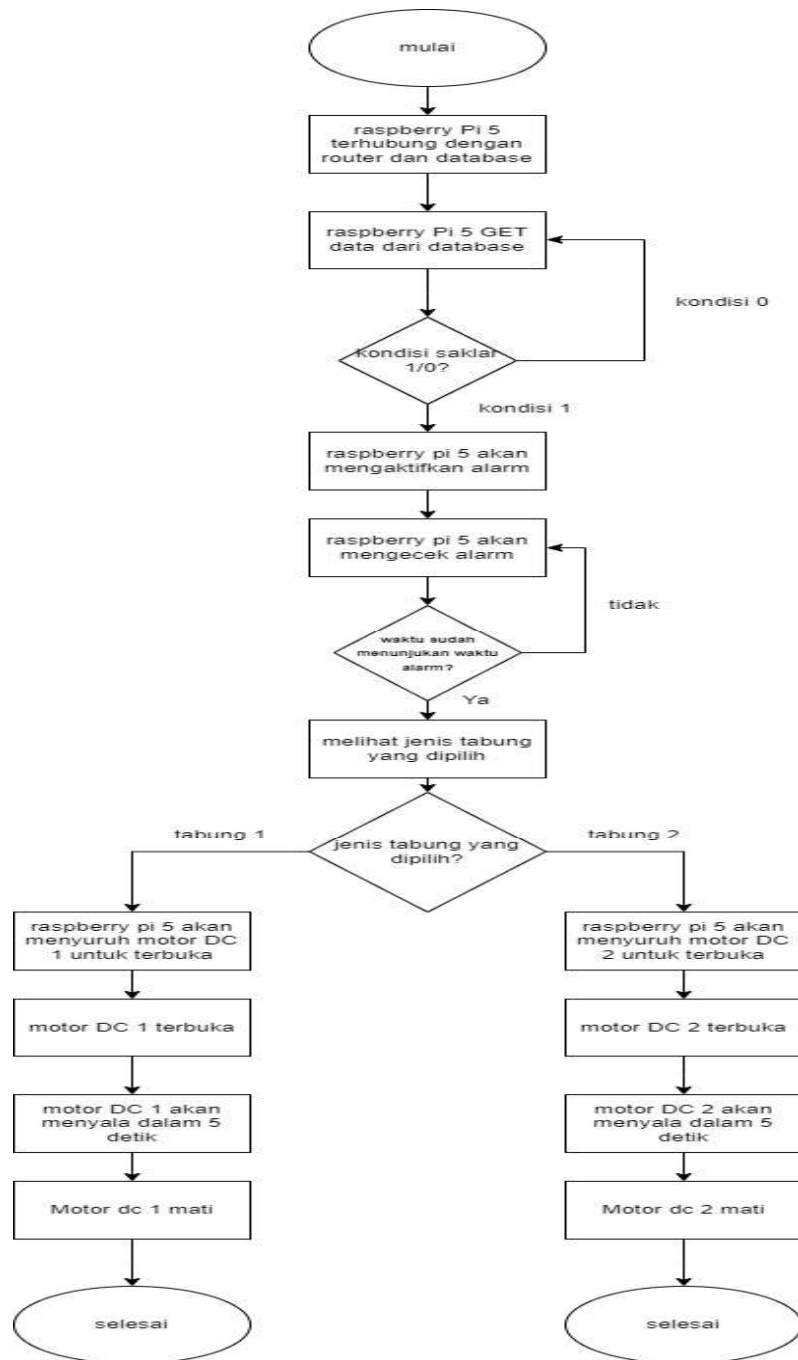
Setelah membuat desain perangkat keras. Dalam penelitian ini terdapat alir kerja pemberian pakan secara manual yang diperlihatkan pada gambar 3.10.

Diawali dengan menyalakan *raspberry Pi 5* dan dipastikan sudah terhubung dengan jaringan dan *database*. Setelah itu, pengguna melakukan permintaan GET data dari *database* dan apabila pilihan pengguna adalah tabung 1 maka terdapat dua opsi yang dapat dipilih oleh pengguna. Jika pengguna memilih opsi “on” maka *raspberry pi 5* akan mengaktifkan motor DC 1 untuk bergerak selama 3 detik untuk membuka tabung. Untuk proses pada tabung 2 memiliki kemiripan prosesnya seperti pada tabung 1.



Gambar 3.10 Diagram alir pemberian pakan secara manual

Selain pemberian pakan secara manual terdapat juga pemberian pakan secara penjadwalan. Pada metode ini mengandalkan pemberian pakan berdasarkan waktu yang telah ditentukan. Untuk diagram alur pemberian pakan secara penjadwalan diperlihatkan pada gambar 3.11.



Gambar 3.11 Diagram alir pemberian pakan secara penjadwalan

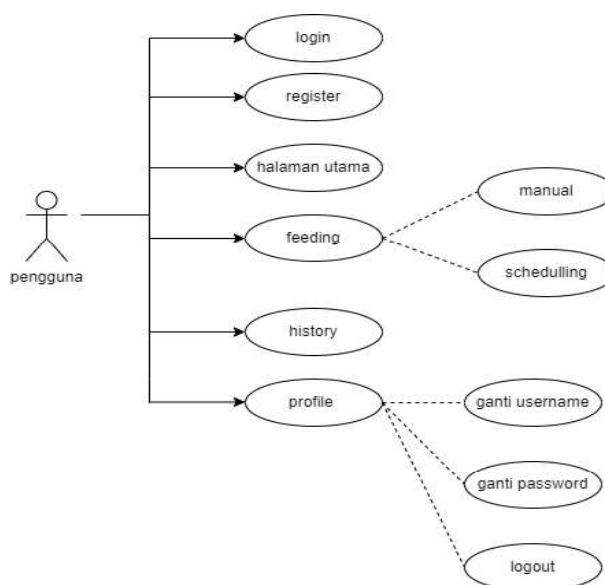
Pada gambar 3.11 diperlihatkan sebuah diagram alir pemberian pakan secara penjadwalan yang dimulai dengan *raspberry pi 5* harus dipastikan terhubung dengan internet dan juga *database*. Setelah itu *raspberry pi 5* akan melakukan permintaan GET data dari *database* untuk memeriksa kondisi tombol saklar, jika bernilai 0 maka *raspberry pi* tidak melakukan tindakan apapun dan melakukan permintaan GET kembali hingga mendapatkan nilai 1. Jika kondisi saklar bernilai

1 maka *raspberry pi* akan menyalakan *alarm* dengan bantuan modul RTC hingga waktu *alarm* yang ditentukan.

Jika waktu yang ditentukan sudah sesuai maka *raspberry pi* 5 melihat jenis tabung yang dipilih, jika belum terdapat jenis tabung *raspberry pi* 5 akan melakukan pengecekan Kembali. Apabila tabung yang dipilih adalah tabung 1 maka *raspberry pi* 5 akan memerintah motor DC 1 untuk terbuka dan motor DC pun menerima perintah tersebut untuk terbuka, lalu motor DC akan menyala selama 5 detik dan setelah itu motor dc akan mati. Begitu pun dengan proses pada tabung 2.

3.3.2 Perancangan perangkat lunak

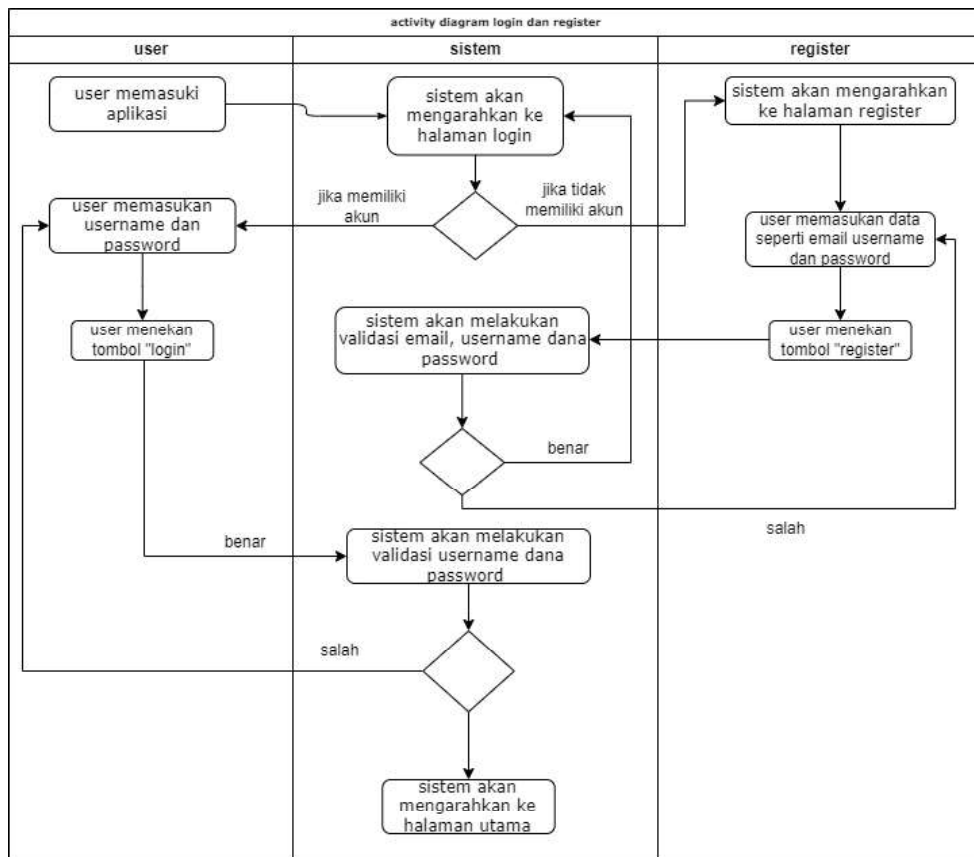
Dalam perancangan perangkat lunak ini terdapat beberapa perancangan diantaranya adalah dengan membuat *use case diagram*, ERD diagram, *activity diagram*, dan perancangan tampilan aplikasi android. Pada gambar 3.12 terdapat *use case diagram*.



Gambar 3.12. *Use case diagram* aplikasi *pet feeder* yang dibuat

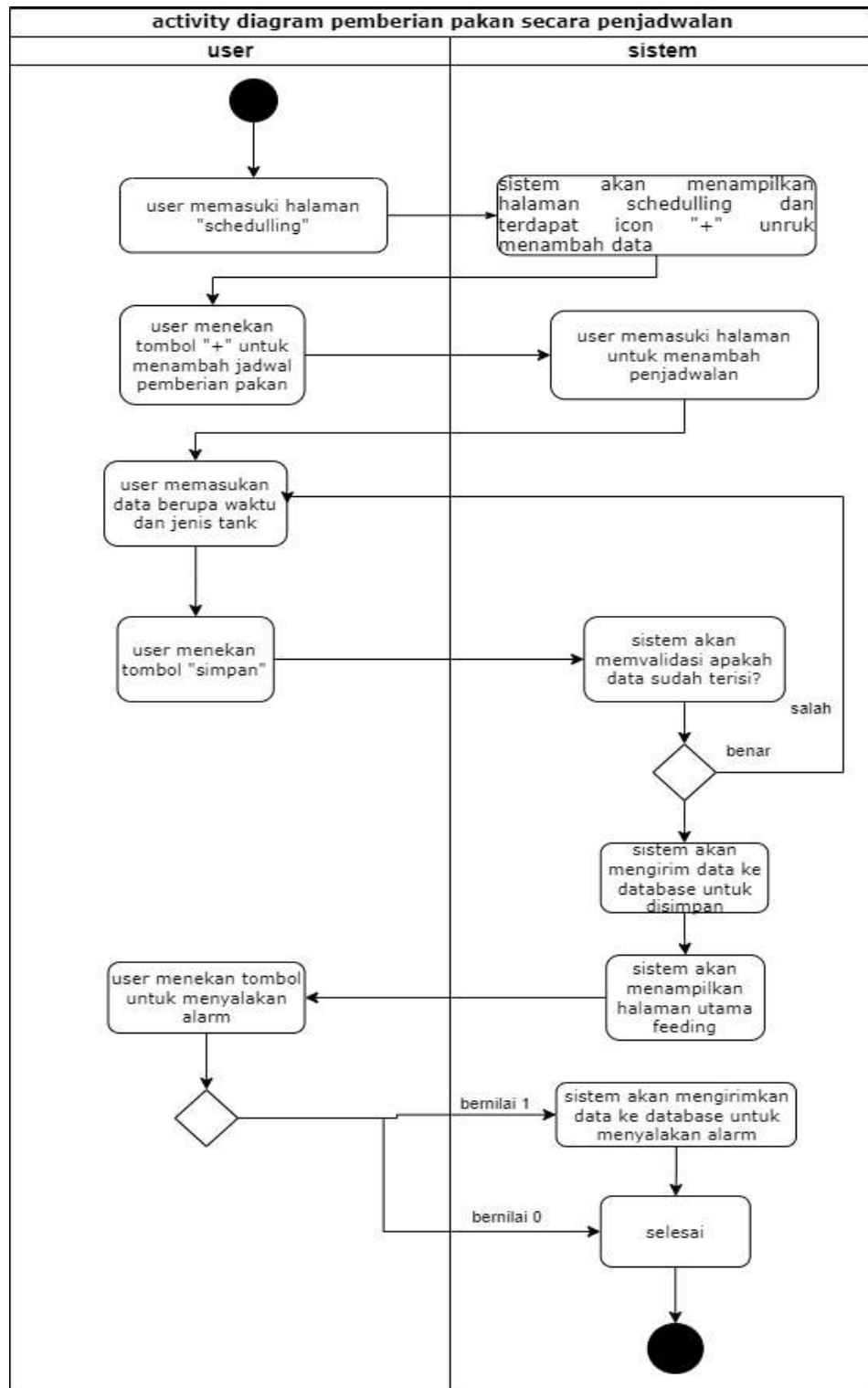
Pada gambar 3.12 diperlihatkan sebuah *use case diagram* yang memiliki fungsi untuk merepresentasikan interaksi dari pengguna dengan sistem yang dibuat, contohnya pengguna dapat mengakses halaman *login*, *register*, halaman utama, halaman *feeding* yang terdiri dari pemberian pakan secara manual dan penjadwalan, *history*, dan *profile* yang didalamnya terdapat fitur ganti *username*, *password*, dan fitur *log out*. Setelah pembuatan *use case diagram* yaitu pembuatan *activity diagram*

yang memiliki fungsi untuk menggambar aktivitas pengguna dalam sebuah system seperti yang diperlihatkan pada gambar 3.13.



Gambar 3.13 Activity diagram login dan register

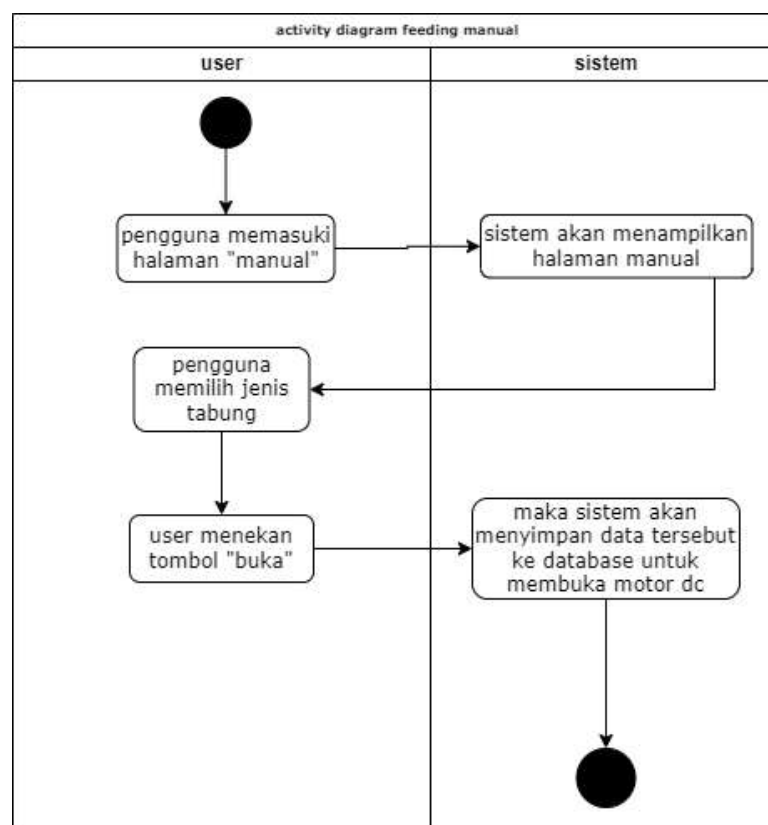
Pada gambar 3.13 diperlihatkan sebuah *activity diagram login dan register* dari perangkat lunak yang dibuat. Fungsi dari *login* bagi pengguna adalah langkah awal sebelum memasuki halaman utama dari aplikasi dengan memasukkan *username* dan *password* serta data pengguna ini akan disimpan untuk menentukan userid nya. Kedua yaitu *register*, disini pengguna melakukan registrasi dengan tujuan jika pengguna yang tidak memiliki akun, hal ini diperlukan untuk menentukan userid pada alat. Selanjutnya terdapat *activity diagram* pada pemberian pakan secara penjadwalan yang dapat dilihat pada gambar 3.14.



Gambar 3.14 *Activity diagram* metode pemberian pakan secara penjadwalan

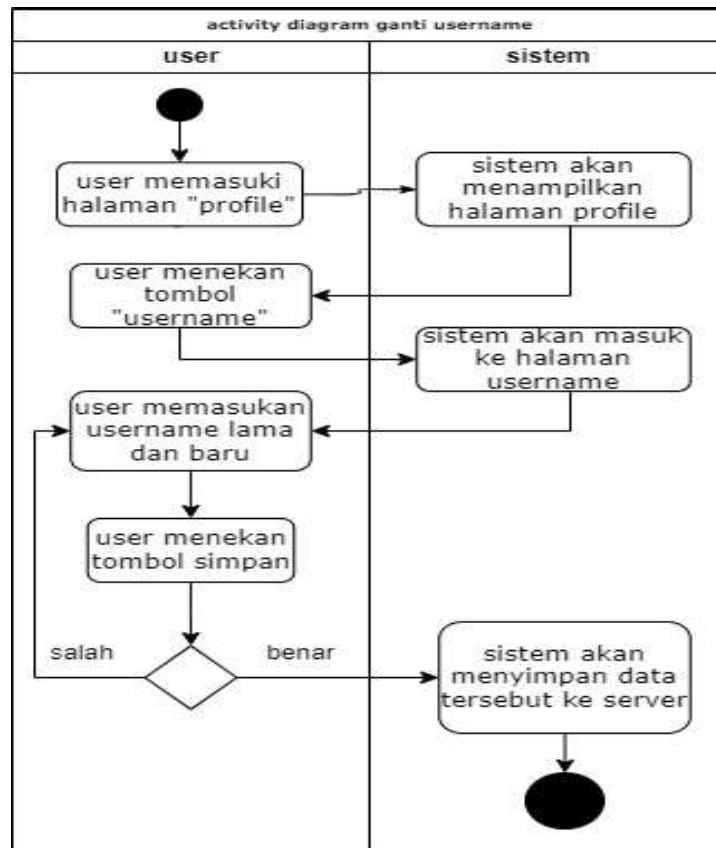
Pada gambar 3.14 diperlihatkan sebuah *activity diagram* metode pemberian pakan secara penjadwalan. Setelah pengguna masuk ke halaman *feeding* dan

memilih halaman penjadwalan maka pengguna diperlihatkan halaman pemberian pakan secara penjadwalan yang didalamnya terdapat kumpulan data pemberian pakan secara penjadwalan dan juga terdapat logo “+“ yang fungsinya adalah menambahkan data penjadwalan baru. Setelah menekan tombol tersebut pengguna harus diharuskan mengisi data seperti memasukan jam dan jenis tabung yang diinginkan. Jika salah satu data tidak terisi maka data tidak akan tersimpan dan jika terisi semua maka sistem akan mengirim data tersebut ke *server* dan pengguna akan dikembalikan ke halaman penjadwalan dan data yang sudah diinputkan sebelumnya akan ditampilkan. Untuk menyalakan penjadwalan tersebut pengguna harus menekan tombol sakelar. Jika bernilai 1 maka sistem akan menyalakan penjadwalan tersebut dan jika bernilai 0 maka akan dimatikan. Maka selesailah pemberian pakan dengan metode secara penjadwalan. Selanjutnya terdapat sebuah *activity diagram* pemberian pakan secara manual yang diperlihatkan pada gambar 3.15.



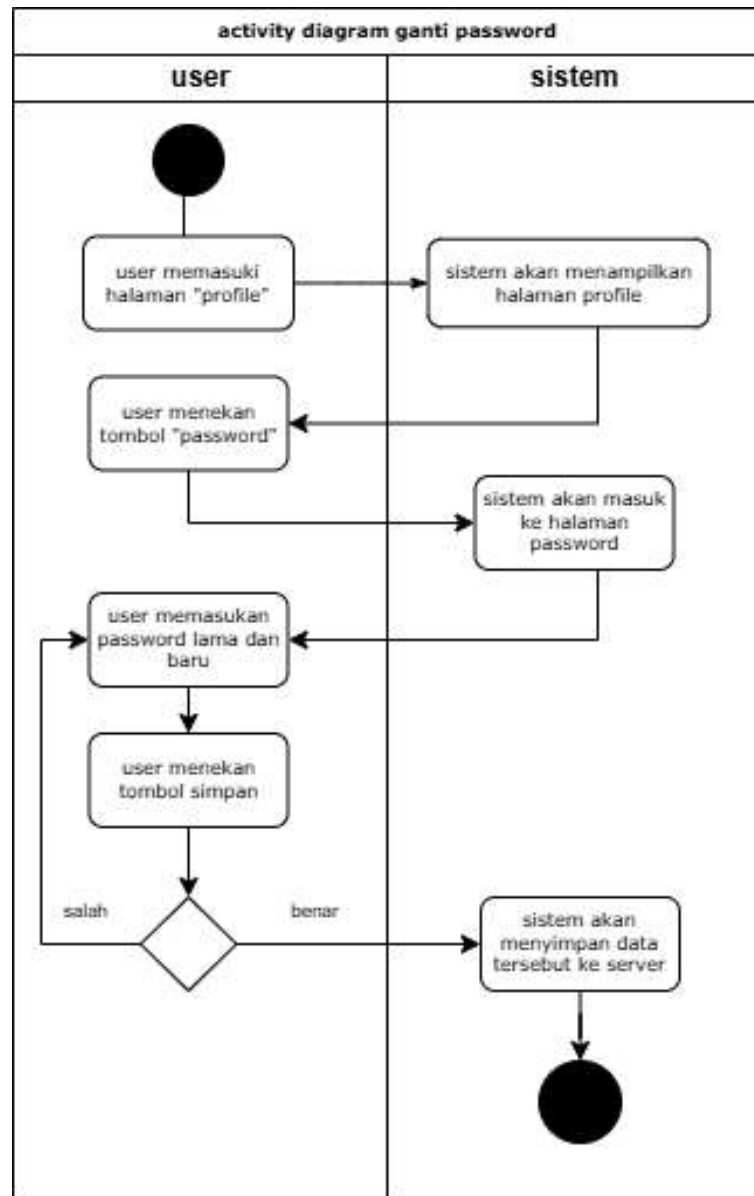
Gambar 3.15 *Activity diagram feeding manual*

Selanjutnya terdapat *activity diagram* untuk mengubah *username* dalam aplikasi yang dirancang. *Activity diagram* tersebut dapat dilihat pada gambar 3.16



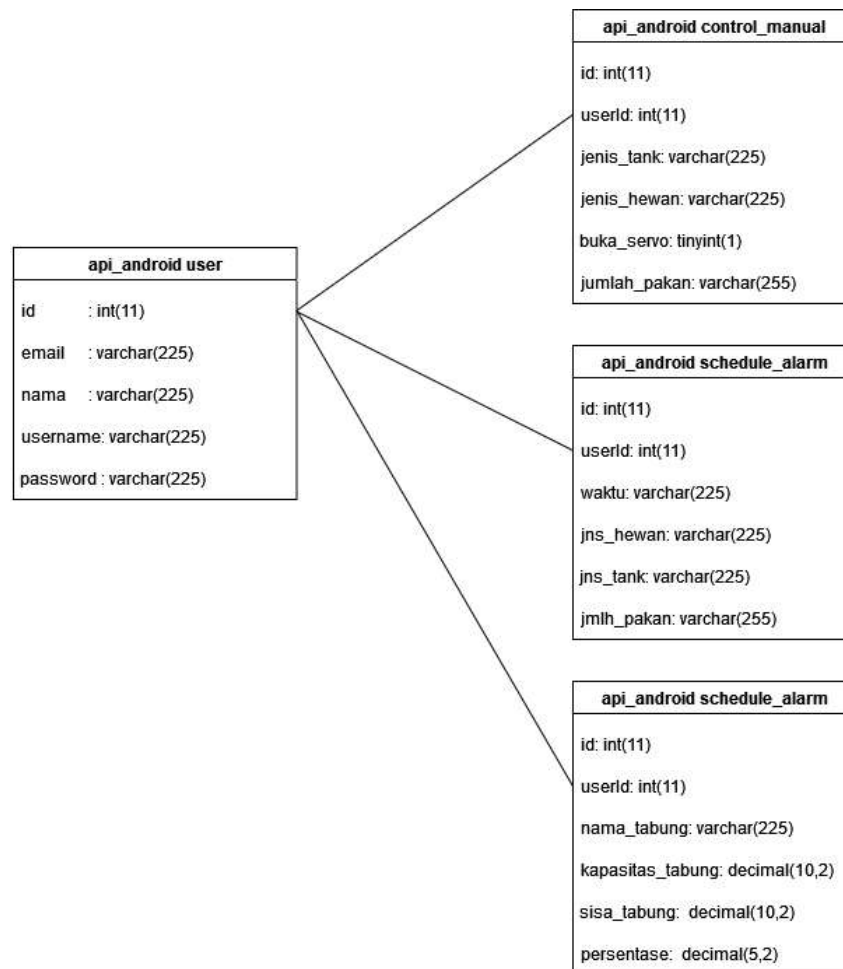
Gambar 3.16 *Activity diagram* mengganti *username*

Pada gambar 3.16 diperlihatkan sebuah *activity diagram* untuk mengganti *username*. Langkah pertama yang harus dilakukan oleh pengguna yaitu memasuki halaman profil dengan menekan tombol “username“. Pada halaman tersebut terdapat beberapa data yang perlu diisi yaitu *username* lama dan *username* baru. Jika benar, maka password terbaru akan disimpan. Selain mengganti *username* pengguna bisa juga melakukan pergantian *password* yang *activity diagram* untuk mengganti *password* yang diperlihatkan pada gambar 3.17



Gambar 3.17 Activity diagram ganti *password*

Pada gambar 3.17 diperlihatkan sebuah *activity diagram* ganti *password* jika pengguna akan mengganti *password*. Langkah pertama yang harus dilakukan oleh pengguna yaitu memasuki halaman profil dengan menekan tombol "password". Pada halaman tersebut terdapat beberapa data yang perlu diisi yaitu *username* lama dan *username* baru. Jika benar, maka *password* terbaru akan disimpan. Selanjutnya terdapat ERD diagram yang diperlihatkan pada gambar 3.18



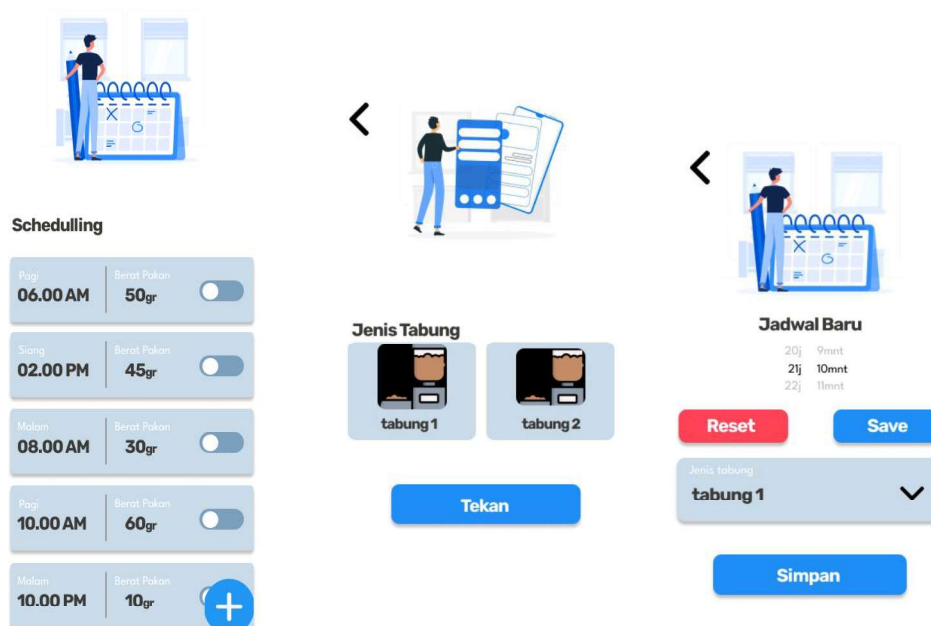
Gambar 3.18 ERD diagram sistem

Diperlihatkan pada gambar 3.18 sebuah ERD (Entity-Relationship Diagram), diagram yang memiliki fungsi untuk merepresentasikan hubungan antar entitas dalam sebuah sistem basis data. Dalam penelitian ini terdapat beberapa tabel diantaranya adalah *user* yang memiliki fungsi menyimpan data pengguna dari id, email, nama, *username*, dan *password*. Lalu terdapat juga *control_manual* yang memiliki fungsi untuk mengontrol keluarnya pakan secara manual yang nantinya akan berpengaruh dalam metode pemberian pakan secara manual dalam tabel ini terdapat beberapa data diantaranya id *userid*, *jenis_tank*, *jenis_hewan*, *buka_servo*, *jumlah_pakan*. Lalu terdapat *schedule_alarm* yang memiliki fungsi menyimpan data dari metode pemberian pakan secara penjadwalan, isi dari tabel ini adalah id, *userid*, *waktu*, *jns_hewan*, *jns_tank*, dan juga *jmlh_pakan*. Terakhir, yaitu terdapat tabel *kapasitas_container* yang memiliki fungsi untuk menyimpan data ketersediaan

pakan pada alat, isi dari tabel ini diantaranya id, userid, nama_tabung, kapasitas_tabung, sisa_tabung, dan persentase.

3.3.3 Desain tampilan android

Pada tahapan pembuatan desain tampilan android dalam penelitian ini menggunakan perangkat lunak figma untuk membuat *mockup* desain dari aplikasi yang akan dibuat. Tujuan dari pembuatan desain tampilan aplikasi android ini adalah memberikan gambaran fitur yang akan dimasukkan dan juga memberikan tampilan yang menarik untuk pengguna. Untuk desainnya sendiri dapat dilihat pada gambar 3.19.



Gambar 3.19 Fitur utama aplikasi

Pada gambar 3.19 dalam desain tampilan tersebut memiliki beberapa fitur utama diantaranya adalah pemberian pakan secara manual dan penjadwalan dengan harapan dari aplikasi ini dapat mengontrol pakan tersebut dari jarak jauh sehingga memudahkan pengguna dalam pemberian pakan.

3.4 Proses pengembangan produk

Pada tahap pengembangan produk yang dibuat ini melalui beberapa proses yaitu dari studi literatur, menentukan komponen yang akan digunakan, menentukan pembaharuan yang akan dilakukan. Pada proses ini ditentukannya perangkat penunjang dari penelitian ini. Pertama penentuan perangkat lunak diantaranya:

1. Visual studio code

2. Arduino IDE
3. Android atudio
4. XAMPP
5. Ngrok
6. Easyeda
7. Sketch up
8. figma

Kemudian untuk perangkat keras yang digunakan dalam penelitian ini diantaranya:

1. Laptop
2. Computer
3. Raspberry PI 5 4GB
4. Esp32 DOIT DEV KIT V1
5. Motor dc
6. Driver motor L298N
7. Sensor loadcell
8. Buzzer
9. Kabel jumper
10. Modul RTC
11. Adaptor 12V

Dalam penelitian ini menekankan pengembangan produk dari pemberian pakan secara manual dan secara penjadwalan. Pada tahap pengembangan ini termasuk juga tahap pengujian dari fungsional alat mulai dari komponen hingga kesatuan alat, aplikasi android, dan API.

3.5 Evaluasi

Setelah melakukan pengembangan pada produk yang dibuat yaitu melakukan evaluasi. Tujuan dari evaluasi ini adalah untuk mengetahui kinerja sistem pet feeder yang pada akhirnya hasil dari evaluasi ini menentukan kelayakan dari pet feeder yang dibuat.

Proses pengujian dilakukan dengan beberapa cara diantaranya adalah untuk aplikasi sendiri menggunakan metode *black box* yang memiliki tujuan untuk memastikan apakah aplikasi yang dibuat sudah berjalan dengan yang dirancang

atau tidak. Untuk perancangan pengujiannya sendiri dapat dilihat pada tabel 3.1, tabel 3.2., tabel 3.3, dan tabel 3.4. Berdasarkan tabel 3.3 dan tabel 3.4 terdapat simbol " – " yang menandakan bahwa tidak adanya data yang diinputkan.

Tabel 3.1 Perancangan pengujian fitur pemberian pakan secara manual

No.	Deskripsi pengujian	Langkah pengujian	Hasil pengujian yang diharapkan
1.	Menguji mengeluarkan pakan dari tabung 1	1. Pengguna memilih tabung 1 2. Pengguna menekan tombol "buka"	Alat dapat mengeluarkan pakan dari tabung 1
2.	Menguji mengeluarkan pakan dari tabung 2	1. Pengguna memilih tabung 2 2. Pengguna menekan tombol "buka"	Alat dapat mengeluarkan pakan dari tabung 2

Tabel 3.2 Perancangan pengujian pada fitur pemberian pakan secara penjadwalan

No.	Deskripsi pengujian	Hasil yang diharapkan
1.	Sistem dapat membuat data penjadwalan baru	Data penjadwalan tersimpan di <i>database</i>
2.	Sistem dapat melakukan pembaharuan pada jadwal yang sudah ada	Data penjadwalan diperbaharui dan tersimpan di <i>database</i>
3.	Sistem dapat melakukan penghapusan data yang sudah ada	Data penjadwalan berhasil dihapus dari <i>database</i>
4.	Penjadwalan yang sudah ada dapat dijalankan	Pakan keluar sesuai dengan tabung yang dipilih ketika waktu penjadwalan tiba

Tabel 3.3 Perancangan pengujian aplikasi

Halaman	Deskripsi pengujian	Input	Ekspektasi pengujian
Login	Pengguna memasukan username dan password untuk masuk ke halaman utama	Username: xboxsteam Password: xboxsteam0123	Pengguna berhasil masuk ke halaman utama
Register	Pengguna menambahkan data baru untuk masuk ke aplikasi	Email:ABC@gmail.com Nama: bambang Username: anjayyotor Password: anjayYotor	Pengguna berhasil menambahkan data dan beralih ke halaman login
Halaman utama	Pengguna ditampilkan halaman utama yang didalamnya terdapat nama user dan progress bar yang isinya ketersediaan pakan	-	Nama pengguna dan ketersediaan sesuai
Halaman feeding	Pengguna memasuki halaman feeding	-	Sistem menampilkan halaman feeding yang didalamnya terdapat scheduling, otomatis, dan manual
Halaman penjadwalan	Pengguna masuk ke halaman penjadwalan	-	Pengguna berhasil memasuki halaman penjadwalan dan menampilkan data penjadwalan (jika ada)
Halaman menambah jadwal baru	Pengguna memasukan data untuk penjadwalan	Waktu: 10:00 PM Jenis tabung: tabung 1	Pengguna berhasil memasukan data dan menampilkan data tersebut dihalaman penjadwalan

Halaman	Deskripsi pengujian	Input	Ekspektasi pengujian
halaman pemberian pakan manual	Pengguna memasuki halaman manual dan melakukan pemberian pakan secara manual	jenis tabung : 1 atau 2 tombol: “buka” atau “tutup”	Pengguna berhasil memberikan pakan kepada hewan dengan menekan tombol “buka” dan berhenti memberikan pakan ketika menekan tombol “tutup”
Halaman history	Pengguna ditampilkan riwayat keluarnya pakan	-	Sistem menampilkan data riwayat pakan
Halaman profile	Pengguna ditampilkan halaman profile yang isinya adalah update username dan update password	-	Sistem menampilkan halaman profile
Halaman ganti username	Pengguna ditampilkan halaman ganti username dan mengganti username	Username lama: anjayyotor Username baru: anjayani	Sistem menyimpan username baru dan ditampilkan di halaman utama
Halaman ganti password	Pengguna ditampilkan halaman ganti password dan mengganti password	password lama: anjayani password baru: anjayYani	Sistem menyimpan password baru
Log out	Pengguna melakukan log out untuk mengakhiri sesi	Tombol: log out	Sistem menuju ke halaman login dan sesi disudahi

Tabel 3.4 Perancangan pengujian API

Endpoint	Deskripsi pengujian	Input	Hasil yang diharapkan
<i>/scheduleAlarm</i>	Mengirimkan permintaan POST ke API	Data seperti <i>userId</i> , waktu, <i>jns_tank</i> , dan <i>kondisi_switch</i> dalam bentuk JSON	Menampilkan pesan “data berhasil dimasukkan” dan data terkirim ke <i>database</i>
<i>/getscheduleAlarm</i>	Mengirimkan permintaan GET ke API	-	API akan menerima pesan dan menampilkan beberapa data seperti <i>id</i> , waktu, <i>jns_hewan</i> , <i>jns_tank</i> , dan <i>jmlh_pakan</i>
<i>/deletedcheduleAlarm</i>	Mengirimkan permintaan DELETE ke API	Data seperti <i>userid</i> dan <i>id</i> dalam bentuk JSON	API akan melakukan penghapusan data
<i>/updatechedulealarm</i>	Mengirimkan permintaan POST ke API	Data seperti <i>id</i> , waktu, <i>jns_tank</i>	API akan melakukan pembaharuan data
<i>/postregister</i>	Mengirimkan permintaan POST ke API	Data seperti email, nama, <i>username</i> , dan <i>password</i>	API akan melakukan penambahan user di <i>database</i>
<i>/login</i>	Mengirimkan permintaan POST ke API	Data seperti <i>username</i> dan <i>password</i>	API akan melakukan sesi

Endpoint	Deskripsi pengujian	Input	Hasil yang diharapkan
			login untuk masuk ke aplikasi
<i>/getuser</i>	Mengirimkan permintaan GET ke API	-	Api akan melakukan mengambil data user dengan data id, email, nama, <i>username</i> , <i>password</i>
<i>/changeusername</i>	Mengirimkan permintaan POST ke API	Data seperti <i>username</i> lama dan <i>username</i> baru	API akan mengganti <i>username</i> dalam database
<i>/changepassword</i>	Mengirimkan permintaan POST ke API	Data seperti <i>password</i> lama dan <i>password</i> baru	Api akan mengganti <i>password</i> lama dalam <u>database</u>
<i>/updateswitch</i>	Mengirimkan permintaan POST ke API	Data seperti id dan kondisi_switch	Api akan mengubah kondisi switch dalam <i>database</i>
<i>/buttonpress</i>	Mengirimkan permintaan POST ke API	Data seperti userid, jenis_tank, buka_servo, dan jumlah_pakan	API akan mengubah kondisi pemberian pakan secara manual
<i>/getdatamanual</i>	Mengirimkan permintaan GET ke API	-	API akan mendapatkan data dari kontrol manual seperti id,

Endpoint	Deskripsi pengujian	Input	Hasil yang diharapkan
			jenis_tank, jenis_hewan, jumlah_pakan untuk halaman riwayat pemberian pakan
<i>/getcontraspi</i>	Mengirimkan permintaan GET ke API	-	Mendapatkan data pemberian pakan secara manual yang diinputkan sebelumnya untuk menentukan jenis tabung mana yang harus dijalankan
<i>/getscheduledata</i>	Mengirimkan permintaan GET ke API	-	Mendapatkan data penjadwalan untuk menyalakan alarm seperti id, waktu, jns_hewan, jns_tank, jmlh_pakan, kondisi_switch
<i>/GETketersediaan</i>	Mengirimkan permintaan GET ke API	-	Mendapatkan data persediaan pakan dari <i>database</i>

Endpoint	Deskripsi pengujian	Input	Hasil yang diharapkan
<i>/kapasitasawal</i>	Mengirimkan permintaan POST ke API	Data seperti id dan kapasitas_tabung yang diinginkan	Dapat memasukan data kapasitas awal dari tabung
<i>/update_sisapakan</i>	Mengirimkan permintaan POST ke API	Data seperti id	Dapat melakukan pembaharuan data ketika ada kondisi tabung terbuka
<i>/get_button</i>	Mengirimkan permintaan GET ke API	-	Mendapatkan kondisi tombol bernilai <i>true</i> atau <i>false</i>
<i>/post_button</i>	Mengirimkan permintaan POST ke API	Data aksi yang bernilai 1 atau 0	Dapat mengubah kondisi tombol fitur otomatis bernilai 0 atau 1

Selain melakukan pengujian secara perangkat lunak dilakukan juga pengujian secara perangkat keras yang tujuannya untuk menghitung rata-rata pakan yang keluar dalam mode pemberian pakan secara manual dan pemberian pakan secara penjadwalan yang memiliki rumus perhitungan yang perlihatkan pada persamaan 3.1.

$$A = \frac{B}{C} \quad (1)$$

Persamaan 3.1 Rumus menghitung rata-rata keluarnya pakan

Keterangan:

A = hasil rata-rata keluarnya pakan

B = total pakan yang keluar

C = total pengujian yang dilakukan